Visual Simulation of Compressible Snow with Friction and Cohesion

Tetsuya Takahashi[1)] **Issei Fujishiro**[2)] **Tomoyuki Nishita**[3)]

1) UEI Research / Graduate School of Science and Technology, Keio University
2) Graduate School of Science and Technology, Keio University
3) UEI Research / Faculty of Economic Information, Hiroshima Shudo University
E-mails: tetsuya.takahashi@uei.co.jp
fuji@ics.keio.ac.jp
tomoyuki.nishita@uei.co.jp

Abstract
Recent advances in physically-based simulation have made it possible to simulate various kinds of natural phenomena. However, characteristic behavior of accumulated snow while being compressed due to pressure has been insufficiently simulated merely with a height field approximation. In this paper, we propose a new method for simulating compressible snow by approximating snow as a set of porous snow particles. We introduce a new parameter called *durability* for each porous snow particle, which represents the rate of undamaged snow structures. The compression of accumulated snow is achieved by absorbing the impact from solids on the snow by taking the durability of snow particles into account. In addition to the compressibility, we incorporate *friction* and *cohesion* into our simulation framework to represent various effects of accumulated snow. For enhanced visual reality, fine scale simulations are performed as a post-processing. Several examples demonstrate the versatility of our method.

1 Introduction

Snow can be seen in various places and deeply linked to our daily lives including disasters, entertainments, and sports in cold regions. Visually appealing and attractive snow scenes have been frequently adopted in applications such as feature films and video games. However, simulating accumulated snow still remains as a challenging issue due to its complexity. For example, the properties of constituent particles of accumulated snow can be easily changed by their own weight and the influence from the surrounding environment, and hence the characteristics of the snow drastically vary according to the state of the particles. Because it is difficult to handle such complicated conditions, three dimensional dynamics of the accumulated snow has been insufficiently simulated merely with a height field approximation.

Since air volume occupies about 90% of the whole volume of freshly-fallen snow according to Nakaya [14], newly accumulated snow collapses releasing air from inside when strongly pressed, and consequently loses the original volume of the snow. Although such compression is one of the most important features of the accumulated snow, fast and simple methods for fully addressing this issue have not yet been proposed.

In this paper, therefore, we propose a new method for simulating three dimensional and compressible accumulated snow. We introduce a new parameter called *durability* for each of porous snow particles, to indicate how much snow structures remain without being damaged, and approximate snow volume as a continuum with these particles. In order to simulate the dynamics of accumulated snow, we employ *Fluid-Implicit-Particle* (FLIP) [22] as our underlying continuum solver since the accumulated snow partly behaves as with inviscid fluid, viscous fluid, or granular materials. We realize the compression of accumulated snow by reducing the effect of the pressure from solids impacting on the snow taking the durability of porous snow particles into account. We also incorporate friction and cohesion, which are important features of accumulated snow, to improve the versatility of our method. Friction of accumulated snow is simulated by employing the Drucker-Prager yield condition and considering local pressures in the snow volume, and cohesion is achieved by the control of a durability-related parameter in the yield condition. We perform fine scale simulations as a post-processing to improve the visual quality of accumulated snow. Figure 1 illustrates important features of our method: compression (left), friction (middle), and cohesion (right).

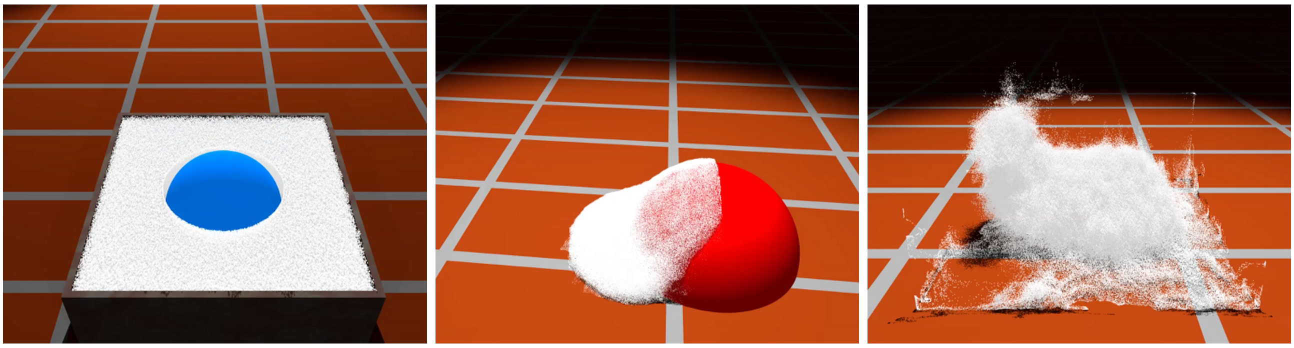


Figure 1: (Left) a block of snow in a gray box compressed without scattering by a solid ball moving along a vertical path. (Middle) a ball-shaped frictional white powder dropped onto a red solid ball. (Right) a bunny-shaped white powder dropped onto the ground in a virtual box, while preserving its original shape due to the cohesion model, making several shadows.

2 Related Work

In Section 2.1, we briefly survey previous attempts to represent snow scenes. Then in Section 2.2, we focus on simulation methods for granular materials, with which our method partly shares the common features.

2.1 Snow

In the field of computer graphics, many methods for representing snow have been developed with primary focus on realistic scenes of accumulated snow. Nishita et al. [16] modeled accumulated snow by making use of metaballs and rendered them taking optical characteristics which cause multiple scattering into account. Another well-known method for simulating accumulated snow was proposed by Fearing [6], who used polygons and particles, and enabled generation of physically-plausible snow scenes by conducting recursive stability tests for the particles on the polygons. Feldman and O'Brien [7] extended the method of Fearing [6] by considering air stream and produced wind-driven accumulated snow scenes. A level set approach was proposed by Hinks and Museth [10] for smoothly accumulated snow. Festenberg and Gumhold [8] employed a statistical snow deposition model using diffusion equations, and successfully generated snow scenes. Tsuda et al. [20] simulated avalanche with snow smoke. Takahashi and Fujishiro [19] enabled the simulation of adhesive snow interacting with solid objects. Stomakhin et al. [18] proposed a new method which uses *Materials Point Method* (MPM) and simulated the dynamics of accumulated snow in a sophisticated manner.

Though there are many well-known methods for simulating accumulated snow, fast and simple methods for sufficiently addressing the compressibility of three dimensional accumulated snow have not yet been proposed.

2.2 Granular Materials

Bell et al. [5] proposed a molecular dynamics method, which is similar to *Discrete Element Method* (DEM), for simulating granular materials. Although their method is versatile and enabled two-way coupling with rigid bodies composed of granular particles, the number of simulation particles is likely to be extremely large, and makes this method impractical.

Li and Moshell [13] proposed a height field method which approximates massive grains in order to reduce the computational cost. Sumner et al. [21] extended this method for generating footprints and tracks, and Onoue and Nishita [17] developed multi-valued height fields and partly simulated three dimensional effects. Although methods using a height field approximation can efficiently capture general motions of granular materials, visually-attractive three dimensional effects are limited.

To sufficiently and efficiently simulate the three dimensional effects of granular materials, a continuum approach was introduced by Zhu and Bridson [22], who simulated sand using FLIP. Narain et al [15] simulated cohesionless granular materials improving the method in [22]. Alduán and Otaduy [1] and Ihmsen et al. [11] generated motions of granular materials in the *Smoothed Particle Hydrodynamics* (SPH) framework.

3 Simulation Framework

We assume that accumulated snow is a continuum since individual snow particles are quite tiny, and thus their very detailed dynamics is visually-ignorable. We herein employ FLIP [22] as our continuum solver for generat-

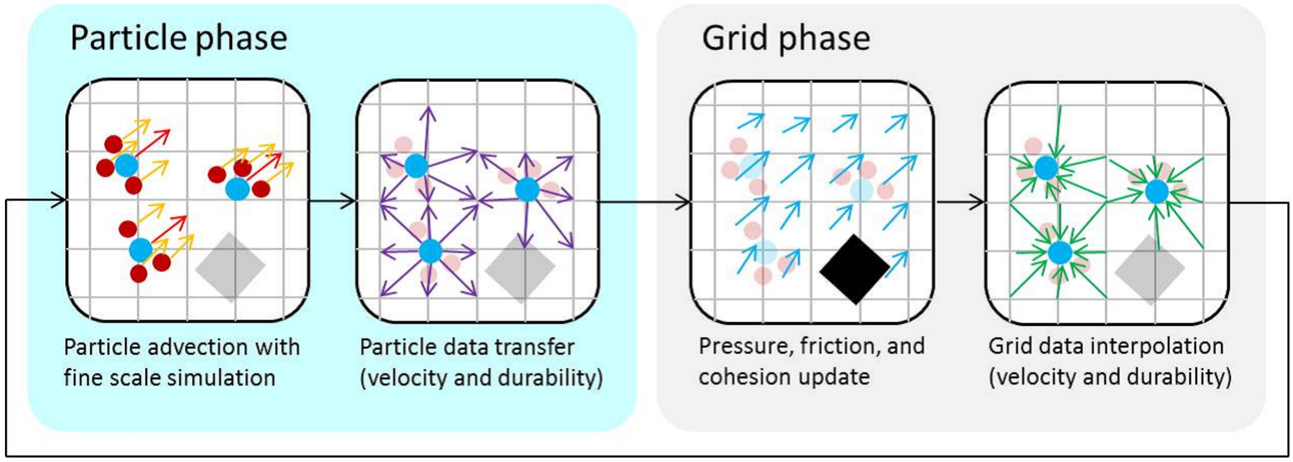


Figure 2: An overview of our method, where gray lines represent a grid, a black box is a solid body, blue particles are porous snow ones, and red particles are fine scale ones. These objects are translucently drawn when they are irrelevant to the simulation in each step. Our simulation has a loop consisting of two phases, each of which has two steps. In the particle phase, particle positions are updated while the fine scale simulation is performed at the first step, and velocity and durability of particles are transferred onto the grid at the second step. In the grid phase, computations of snow motions are conducted by updating pressure, friction, and cohesion at the first step, and grid velocity and durability are interpolated to the particles with FLIP at the second step.

ing snow motions because the solver is robust regardless of particle distributions unlike SPH and DEM, and this feature is necessary for our goal. Figure 2 illustrates an overview of our method. Our method consists of two phases (particle phase and grid phase), each of which has two steps, and the phases are executed alternately and repeatedly.

3.1 Fundamentals

The motion of a continuum is represented with density ρ and velocity $\mathbf{u}$. If we denote the Lagrangian derivative by $D/Dt = \partial/\partial t + (\mathbf{u} \cdot \nabla)$, Navier-Stokes equations are given as

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla p + \nabla \cdot \mathbf{s} + \mathbf{f}_{\text{ext}}, \tag{1}$$

where p denotes pressure, $\mathbf{s}$ deviatoric or frictional stress, and $\mathbf{f}_{\text{ext}}$ external forces. We obtain the dynamics of the continuum by solving Eq. (1) with Dirichlet (for free surfaces) and Neumann (for solid objects) boundary conditions, and the incompressibility constraint

$$\nabla \cdot \mathbf{u} = 0.$$

3.2 FLIP Solver

In FLIP [22], particle properties are transferred onto the grid. Let A denote an arbitrary physical quantity except density, m mass of a particle, and W a kernel. We employ an SPH-like interpolation as in [4, 2], and A and ρ at position $\mathbf{x}$ are transferred as follows:

$$A(\mathbf{x}) = \frac{\sum_i m_i A_i W_{\text{sharp}}(\mathbf{x}_i - \mathbf{x}, h)}{\sum_i m_i W_{\text{sharp}}(\mathbf{x}_i - \mathbf{x}, h)}, \qquad \rho(\mathbf{x}) = \sum_i m_i W_{\text{smooth}}(\mathbf{x}_i - \mathbf{x}, h), \tag{2}$$

where i denotes the index of a particle within the effective radius h. With a vector $\mathbf{r}$, we use the following kernels proposed in [2] because of their simplicity and efficacy:

$$W_{\text{sharp}}(\mathbf{r}, h) = \begin{cases} h^2/||\mathbf{r}||^2 - 1 & 0 \le ||\mathbf{r}|| \le h, \\ 0 & \text{otherwise}, \end{cases}$$

$$W_{\text{smooth}}(\mathbf{r}, h) = \begin{cases} 1 - ||\mathbf{r}||^2/h^2 & 0 \le ||\mathbf{r}|| \le h, \\ 0 & \text{otherwise}. \end{cases}$$

We employ the ghost fluid method [9] and variational framework [3] for accurate simulations.

3.3 Fine Scale Simulation

In our method, since simulation particles represent clamp of matter or distributions of a continuum instead of individual snow particles, we do not directly render these simulation particles. For particles used in rendering and enhanced visual reality, we perform a fine scale simulation as a post-processing [11].

In the fine scale simulation, we ignore interactions among particles for efficiency. Fine scale particles are

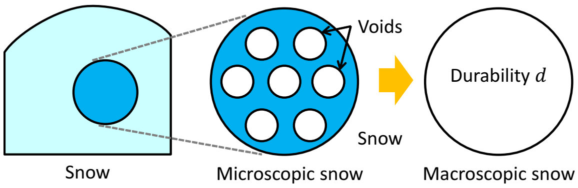


Figure 3: Definition of a porous snow particle.

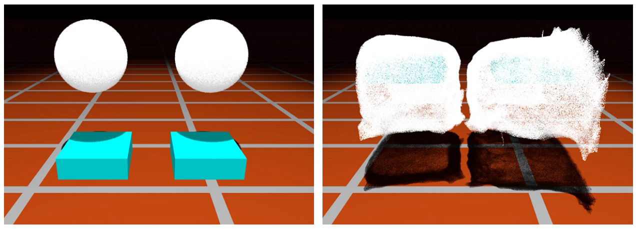


Figure 4: Snow balls with different durabilities interacting with moving cuboids.

governed by the dynamics of the continuum at high density points, and can freely disperse due to external forces in low density areas. Hence, we compute the velocity of fine scale particles $\mathbf{v}$ by taking external forces $\mathbf{f}_{\text{ext}}$ into account:

$$\mathbf{v} = \alpha\mathbf{u} + (1-\alpha)(\mathbf{u} + \mathbf{f}_{\text{ext}}\Delta t/m),$$
$$\alpha = \begin{cases} \rho/\rho^* & \rho \leq \rho^*, \\ 1 & \text{otherwise}, \end{cases}$$

where Δt denotes a time step and ρ^* a threshold density.

4 Compressibility

Snow has compressibility, due to air voids in its volume, which is not exhibited by inviscid fluid, viscous fluid, nor granular materials. Section 4.1 describes how to handle these voids in our model, and Section 4.2 explains the effects of colliding objects to realize the compression of accumulated snow.

4.1 Air Voids

Most part of a snow volume is occupied by air since it has myriad air voids in a sampled volume unlike other continua. Accumulated snow, therefore, possesses a characteristic trait that it loses the volume releasing air stored inside when being pressed. To simulate this phenomenon, we utilize porous snow particles, which were inspired by the method in [12]. As illustrated in Figure 3, we handle snow particles in a macroscopic scale by parameterizing the detailed air voids and microscopic snow structures as a scalar variable, *durability* d $(0 \leq d \leq 1)$, which indicates how much the original snow structures remain without damage. Note that durability $d = 1$ means that snow structures of a particle are intact.

4.2 Effects of Colliding Objects

When a solid object collides with accumulated snow, the snow absorbs the impact owing to its voids while snow structures are damaged. Therefore, the velocity of the snow volume is not likely to be fully affected by the pressure from the solid object due to the voids acting as buffers. Consequently, the velocities of the snow volume and the solid object are not the same, and the non-zero relative velocities make the snow get compressed.

To realize the behavior of compressible accumulated snow, we transfer the durability of particles to the grid using Eq. (2), and for each grid cell, compute the velocity transfer factor b $(0 \leq b \leq 1)$, which controls the extent of influence of solid velocity to the continuum motions by

$$b = b_0 - k_b(d_0 - d),$$

where b_0 denotes an initial velocity transfer factor, $k_b (0 \leq k_b)$ a coefficient of velocity transfer factor, and d_0 an initial durability. Then, a Poisson equation is modified as

$$\frac{\Delta t}{\rho}\nabla^2 p = \nabla \cdot (\mathbf{u} - b\mathbf{u}_s),$$

where $\mathbf{u}_s$ denotes the solid velocity. Figure 4 illustrates the effect of durability with an example of two snow balls interacting with two moving cuboids, respectively. The left snow ball with high durability absorbs the impact due to the left cuboid, whereas the right one with low durability is strongly scattered.

Since when accumulated snow pressed by an object, the structures of the snow are damaged. We assume that the entire energy generated by the collision is used for the destruction of snow structures ignoring the dissipation of the energy as sound and heat for simplicity, and then reduce the grid durability d by

$$d \leftarrow d - k_q p, \tag{3}$$

where k_q denotes a durability change coefficient. Since damaged structures never recover, the durability reduces monotonously. After all the durability values on the grid are updated with Eq. (3), they are interpolated with FLIP [22]. Note that *Particle-In-Cell* (PIC) [22] is inapplicable to the interpolation due to the fast numerical dissipation.

5 Friction and Cohesion

As in the simulations for granular materials, a yield condition is responsible for determining the dynamics of accumulated snow. We employ the Drucker-Prager yield condition with a cohesion term to check whether $\mathbf{s}$ is large enough to make the snow plastically flow:

$$||\mathbf{s}||_F \leq \sqrt{3}\alpha p + c,$$

where $||\mathbf{s}||_F = \sqrt{\Sigma s_{ij}^2}$ denotes the Frobenius norm of $\mathbf{s}$, $\alpha = \sqrt{2/3}\ \sin\theta$, where θ is an angle of repose, and c a cohesion criterion. If the yield condition does not hold in a cell, the cell is labeled as *plastic*, otherwise *rigid*.

If the label of cells is *plastic*, the continuum in the cells plastically flows dissipating its energy owing to the friction. To simulate such behavior, we formulate implicit systems for friction, which are similar to the implicit systems for viscosity proposed by Batty and Bridson [4]. Since friction forces become stronger as pressure gets higher, we compute a friction coefficient μ by

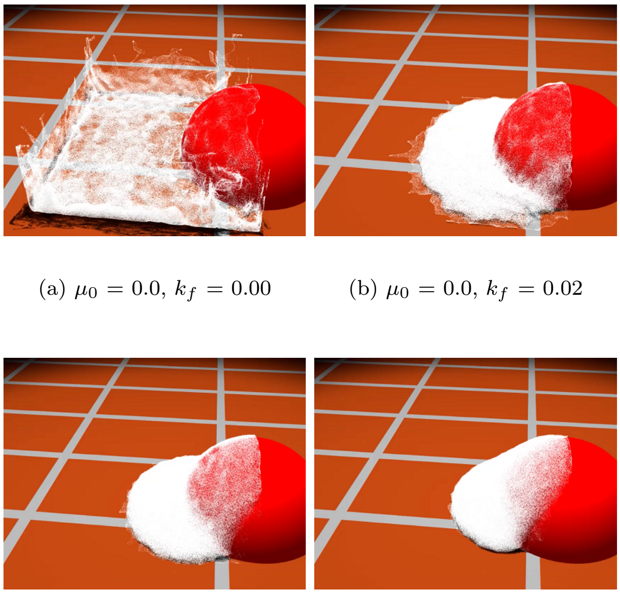
$$\mu = \mu_0 + k_f p,$$

where μ_0 denotes an initial friction coefficient, and k_f a friction change coefficient. Figure 5 demonstrates the effect of our friction model using ball-shaped white powders with different parameter values, dropped onto a red solid ball.

In order to handle cells labeled as *rigid*, we follow the method in [22] for cohesive motions. First, we connect rigid-labeled cells to construct rigid clusters, and compute the linear and angular velocity of the clusters preserving their linear and angular momentum. Then, we update the velocity of rigid-labeled cells with the linear and angular velocity of the rigid clusters. Accumulated snow generally shows different cohesion depending on positions in the snow volume. For example, if the structures of accumulated snow are damaged, the snow shows higher cohesion, while freshly-fallen and undamaged snow does not exhibit an adhesive property. By taking these facts into account, we compute the cohesion criterion c with the durability d by

$$c = c_0 + k_c(d_0 - d),$$

where c_0 denotes an initial cohesion criterion, and k_c a cohesion change coefficient. Figure 6 illustrates the difference of a dropped bunny with different conditions. In this scene, while the bunny with little cohesion scatters as with fluid, the highly cohesive bunnies preserve their original shapes.



(a) $\mu_0 = 0.0$, $k_f = 0.00$ (b) $\mu_0 = 0.0$, $k_f = 0.02$

(c) $\mu_0 = 0.0$, $k_f = 0.06$ (d) $\mu_0 = 0.0$, $k_f = 0.15$

Figure 5: A comparison of friction effects with different parameter values.

Table 1: Statistics of all the scenes.

Figure number	Grid size	Particles Simulation/fine scale	Time/ simulation step (s)
4	50^3	28.3k/1.7M	2.1
5	50^3	14.2k/0.5M	0.6
6	60^3	64.8k/1.9M	3.3
7	40^3	128.3k/0	0.8
8	50^3	224.1k/4.0M	1.4
11	50^3	33.0k/2.0M	1.1

6 Results and Discussions

We implemented our method in C++ and parallelized it employing OpenMP 2.0. All the scenes were executed on a PC with Intel Core i7 3.50 GHz and RAM 16.0 GB. We used six threads in total and obtained the accelerated simulation performance by a factor from two to four. All the results were rendered with a raytracer POV-Ray 3.7. We rendered particles used in the fine scale simulation and needed about 90 seconds per frame on average for off-line rendering of Figure 1 (left). We tabulate the simulation conditions and performances of all the results in Table 1.

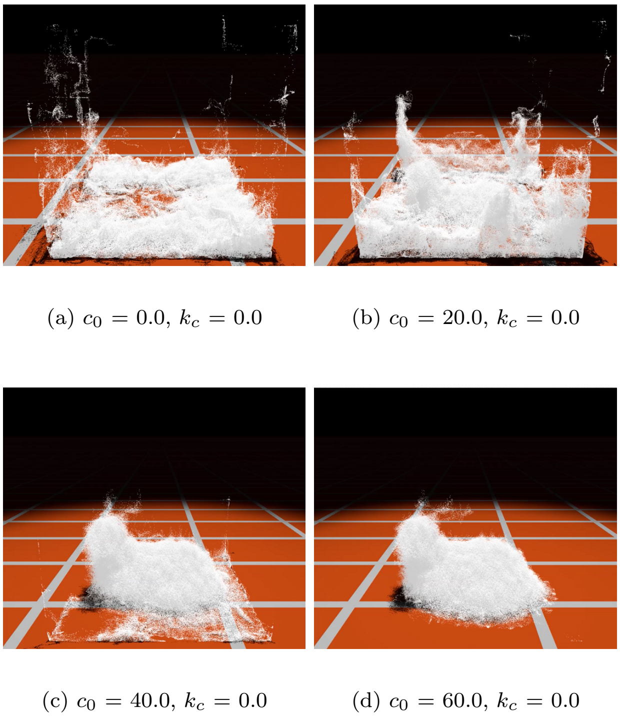


Figure 6: A comparison of cohesion effects with different parameter values.

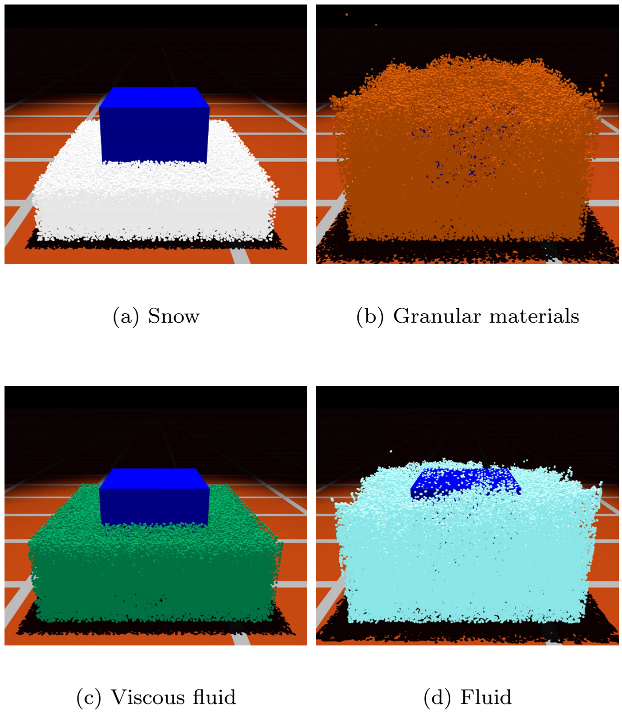


Figure 7: Motion comparisons of continua interacting with a solid cube moving along a vertical path, where simulation particles are rendered for visualization.

6.1 Results

Figure 7 compares snow (a), granular materials (b), viscous fluid (c), and fluid (d) with particle representations, where particles are compressed by the cube moving along a vertical path. While the snow block is compressed due to the pressure from the cube without scattering and flowing in Figure 7 (a), the granular materials are scattered by the cube in Figure 7 (b). In Figure 7 (c), the viscous material is pushed by the cube and rises avoiding the cube, and the fluid flows due to the pressure from the cube in Figure 7 (d).

Figure 8 illustrates a snow block being compressed by the solid ball, and Figure 9 provides side views of the scene (Figure 8) in terms of density to elucidate the snow compression, where the front side is clipped for visualization, and particles are colored according to their density (low density: white and high one: red). We can observe that the block of snow is compressed without flowing and scattering due to our model of the compressibility, friction, and cohesion, and that many particles are gathered under the solid due to the compression. Figure 10 plots a sequence of timings to generate the results in Figure 8. On average, the computational time consumed for the simulation is divided as follows: 7% for compressibility, 14% for pressure, 25% for friction, 1% for cohesion, 35% for fine scale simulation, and 18% for the rest. Most of the simulation time is used for the computations of friction and fine scale simulation, and additional cost for the compressibility in our method is relatively low.

Figure 11 (left) demonstrates a snow ball dropped on the solid cube, where the snow stays on the box thanks to the friction and cohesion forces. A corresponding simulation particle representation is shown in Figure 11 (right).

6.2 Discussions

Unlike FLIP [22] that we used, it would be difficult to capture the behavior of compressible snow using a particle-based method such as SPH or DEM. It is because particle distributions in the particle-based methods can have much impact on the motions of the particles, and the particles could cause extremely strong repulsion forces to resolve penetrations and collisions of the particles when compressed by excessively heavy objects. Although adopting very small time steps or merge/split operations of particles would alleviate the problem, a significant increase of computational cost would be anticipated. On the other hand, FLIP [22] is

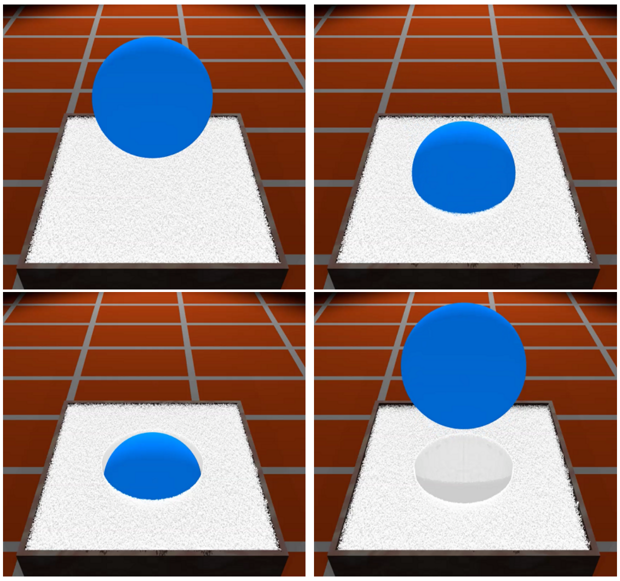


Figure 8: A block of snow compressed by a solid ball moving along a vertical path. Simulation steps are 0, 18, 36, and 132.

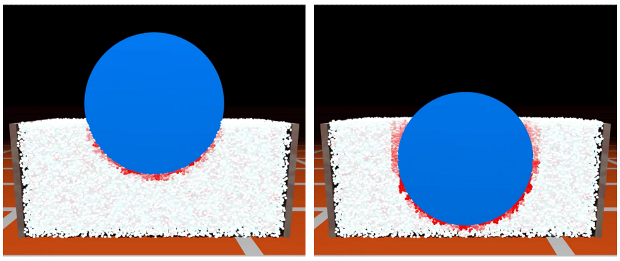


Figure 9: Side views of the snow compression showed in Figure 8, where the front side is clipped for visualization, and particles are colored according to their density (low density: white and high one: red).

fairly tolerable to biased distributions of particles since FLIP [22] does not need to resolve particle penetrations and collisions.

7 Conclusions and Future Work

We have presented a new method for simulating three dimensional and compressible accumulated snow. Our method utilizes porous snow particles, which have *durability*, to allow for the compression of accumulated snow by the control of the impact of solids on the snow. Additionally, we modeled frictional motions of accumulated snow formulating implicit systems, and realized cohesion of snow by taking the durability into account.

In our method, however, accumulated snow cannot

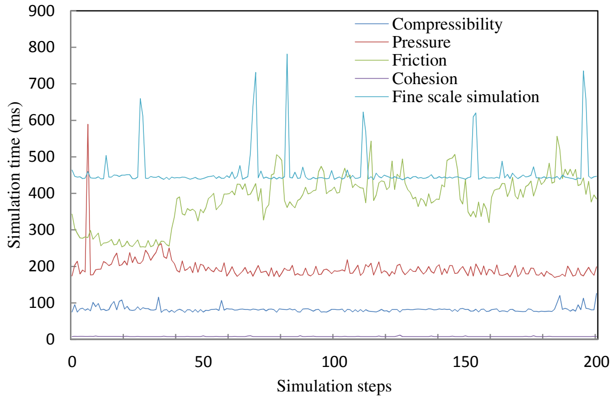


Figure 10: A timing profile of the snow compression in Figure 8.

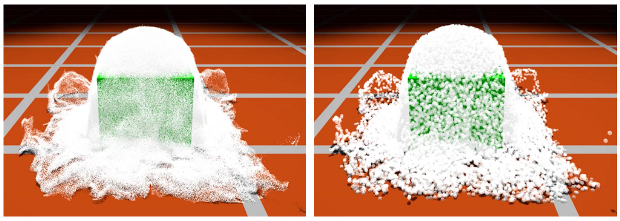


Figure 11: (Left) a snow ball dropped onto a box. (Right) the corresponding simulation particle representation.

freely disperse and gradually flows as with viscous fluid since we assume that a continuum which approximates the snow is incompressible. Narain et al. [15] modified the incompressible FLIP and proposed the unilateral incompressibility constraint to simulate freely-flowing granular materials by solving the linear complementarity problem. Stomakhin et al. [18] utilized MPM to simulate the dynamics of accumulated snow, and produced scattering snow particles. Although our method focused on simple models for fast simulations, these complicated methods would be promising to realize dispersing particles in our framework. In addition, we would like to simulate the phase transition of accumulated snow caused by pressure melting, and render snow particles taking optical properties into account.

References

[1] Alduán, I. and Otaduy, M. A.: "SPH Granular Flow with Friction and Cohesion," in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation,* pp. 25–32, 2011.

[2] Ando, R., Thürey, N., and Turuno, R.: "Pre-

serving Fluid Sheets with Adaptively Sampled Anisotropic Particles," *IEEE Transactions on Visualization and Computer Graphics,* Vol. 18, No. 8, pp. 1202–1214, 2012.

[3] Batty, C., Bertails, F., and Bridson, R.: "A Fast Variational Framework for Accurate Solid-Fluid Coupling," *ACM Transactions on Graphics,* Vol. 26, No. 3, Article 100, 2007.

[4] Batty, C. and Bridson, R.: "Accurate Viscous Free Surfaces for Buckling, Coiling, and Rotating Liquids," in *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation,* pp. 219–228, 2008.

[5] Bell, N., Yu, Y., and Mucha, P. J.: "Particle-Based Simulation of Granular Materials," in *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation,* pp. 77–86, 2005.

[6] Fearing, P.: "Computer Modelling of Fallen Snow," in *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer Graphics and Interactive Techniques,* pp. 37–46, 2000.

[7] Feldman, B. E and O'Brien, J. F.: "Modeling the Accumulation of Wind-Driven Snow," in *SIGGRAPH '02: ACM SIGGRAPH 2002 Conference Abstracts and Applications,* pp. 218–218, 2002.

[8] Festenberg, N. V. and Gumhold, S.: "Diffusion-Based Snow Cover Generation," *Computer Graphics Forum,* Vol. 30, No. 6, pp. 1837–1849, 2011.

[9] Gibou, F., Fedkiw, R., Cheng, L.-T., and Kang, M.: "A Second-Order-Accurate Symmetric Discretization of the Poisson Equation on Irregular Domains," *Journal of Computational Physics,* Vol. 176, No. 1, pp. 205–227, 2002.

[10] Hinks, T. and Museth, K.: "Wind-Driven Snow Buildup Using a Level Set Approach," in *Proceedings of Eurographics Ireland Workshop Series,* Vol. 9, pp. 19–26, 2009.

[11] Ihmsen, M., Wahl, A., and Teschner, M.: "High-Resolution Simulation of Granular Material with SPH," in *Proceedings of the VRIPHYS,* pp. 53–60, 2012.

[12] Lenaerts, T., Adams, B., and Dutré, P.: "Porous Flow in Particle-Based Fluid Simulations," *ACM Transactions on Graphics,* Vol. 27, No. 3, Article 49, 2008.

[13] Li, X. and Moshell, J. M.: "Modeling Soil: Realtime Dynamics Models for Soil Slippage and Manipulation," in *Proceedings of SIGGRAPH,* pp. 361–368, 1993.

[14] Nakaya, U.: *Snow Crystals: Natural and Artificial,* Harvard University Press, 1954.

[15] Narain, R., Golas, A., and Lin, M. C.: "Free-Flowing Granular Materials with Two-Way Solid Coupling," *ACM Transactions on Graphics,* Vol. 29, No. 6, Article 173, 2010.

[16] Nishita, T., Iwasaki, H., Dobashi, Y., and Nakamae, E.: "A Modeling and Rendering Method for Snow by Using Metaballs," *Computer Graphics Forum,* Vol. 16, No. 3, pp. 357–364, 1997.

[17] Onoue, K. and Nishita, T.: "Virtual Sandbox," in *Proceedings of IEEE 2003 Pacific Conference on Computer Graphics and Applications,* pp. 252–259, 2003.

[18] Stomakhin, A., Schroeder, C., Chai, L., Teran, J., and Selle, Andrew.: "A Material Point Method for Snow Simulation," *ACM Transactions on Graphics* Vol. 32, No. 4, pp. 102:1–102:10, 2013.

[19] Takahashi, T. and Fujishiro, I.: "Particle-Based Simulation of Snow Trampling Taking Sintering Effect into Account," in *SIGGRAPH 2012 Posters,* Article No. 7, 2012.

[20] Tsuda, Y., Yue, Y., Dobashi, Y., and Nishita, T.: "Visual Simulation of Mixed-Motion Avalanches with Interactions Between Snow Layers," *The Visual Computer,* Vol. 26, No. 6–8, pp. 883–891, 2010.

[21] Sumner, R. W., O'Brien, J. F., and Hodgins, J. K.: "Animating Sand, Mud, and Snow," *Computer Graphics Forum,* Vol. 18, No. 1, pp. 17–26, 1999.

[22] Zhu, Y. and Bridson, R.: "Animating Sand as a Fluid," *ACM Transactions on Graphics,* Vol. 24, No. 3, pp. 965–972, 2005.